

Structure Capsule Specification

Purpose

This Capsule provides a starting point for applications that need to handle tree or network structures. It will significantly increase the functionality of other capsules.

Functionality

The structure capsule supports the object model depicted in Appendix A. The capsule provides tools or methods for:

- Installing a federated database with a preloaded schema and at least one template database.
- Creating, updating and deleting (optionally) named, hashable and/or indexed instances of the Component object class. It includes the ability to link new instances to other instances to model a structure (or network) and to optionally create alternatives for individual instances.
- The methods will use a sample placement model (segmenting the data into databases, containers and object clusters).
- Finding Component instances by name or key and querying or iterating over all instances of Components. The latter methods are automatically generated during schema definition.
- **Transitive closure:** Multi-hop traversal of the two pre-defined many-to-many associations.
- **Loop avoidance:** Optionally avoiding loops when defining many-to-many associations. This involves doing a transitive closure to see if there is a route from object B to object A before connecting A to B. It is optional because it can significantly slow down structure creation. However, not doing it can lead to loops while trying to navigate the structure.
- **Path finding:** It will leverage the Connection Explorer capsule to find links between nodes.
- **Deep copy:** The ability to copy or move a designated structure to another container.
- *To Be Determined:* A GUI, such as an ooAssist plugin, supporting the above functionality.
- There will be no Objectivity-proprietary configuration management (versioning etc.) capability, though this may be provided in a separate capsule.

Platforms and Languages

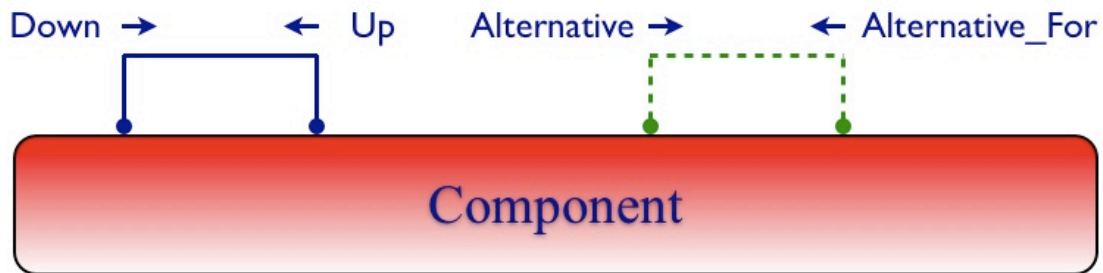
- Windows and Linux.
- C++, Java and C# (later).

Suggested Pricing

- Free and open source, but requiring standard Objectivity/DB licenses to be useful.

Appendix A – Object Model for the Structure Capsule

Object Model for the Structure Capsule



- “Component” is a term for a constituent of a structure.
- A “Component” instance might be:
 - A product, sub-assembly or part in a mechanical design.
 - A cabinet, rack, board or chip in a piece of telecom equipment.
 - An organization, team, person or role in an HR or security system.
 - A unit, vessel, battle group or fleet in a Naval system.
 - A clip, scene or frame in a video production
 - A volume, document, chapter, paragraph, sentence or word in a text processing system.
 - A node in a telecom or IP network.
- The “Alternative/Alternative_For” association is optional.